

**TRABAJO 2: DIAGRAMA DE CLASES PROYECTO “RESCATANDO  
MASCOTAS”**



**CRISTINA SAMBONI SANDOVAL**

**YEISON ALBEIRO RUIZ**

**KEVIN RODRIGUEZ**

**Trabajo**

**Trabajo presentado a ING: GUSTAVO SALAZAR ESCOBAR**

**ADSO PROGRAMACIÓN**

**SENA**

**POPAYÁN**

**JUNIO 22 DE 2025**

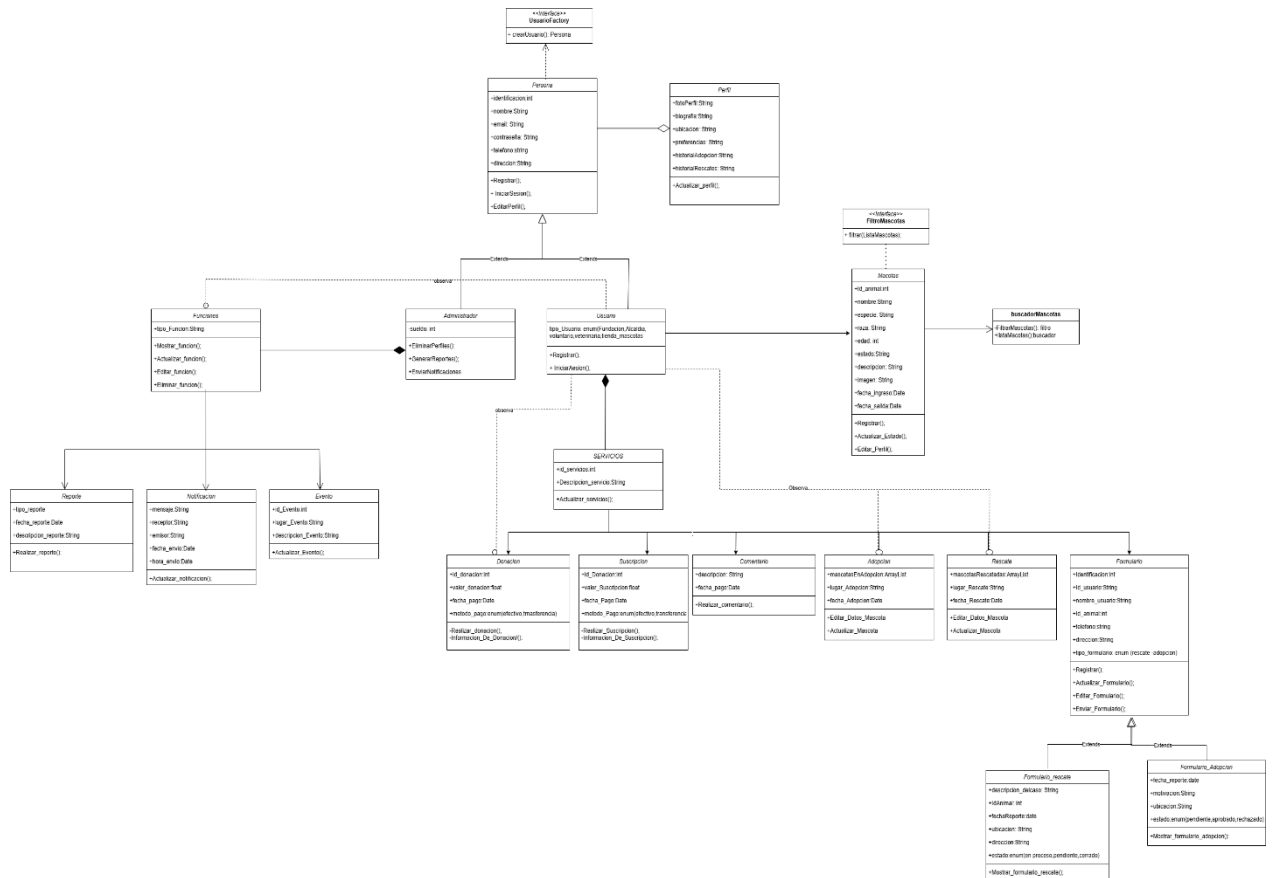
## INTRODUCCION

Este trabajo consiste en realizar el diagrama de clase del proyecto "RESACATANDO MASCOTAS FOREVER", el cual se basa en la temática de UML, logrando plasmar cada una de las clases necesarios para construir la aplicación móvil y web del proyecto.

A través de las consultas en internet y libros de UML se analiza la temática y se estructura un modelo inicial para empezar la construcción del modelado, el cual nos permite revisar situaciones complejas que permitan llegar a soluciones reales.

Este documento presenta el diseño del modelo conceptual orientado a objetos para el sistema "Rescatando Mascotas Forever". A partir de las historias de usuario, se identificaron las clases principales, sus relaciones y funcionalidades, organizadas en un diagrama de clases general y diagramas modulares por funcionalidad. El objetivo es representar de forma estructurada los componentes del sistema siguiendo buenas prácticas de diseño orientado a objetos.

## DIAGRAMA DE CLASE



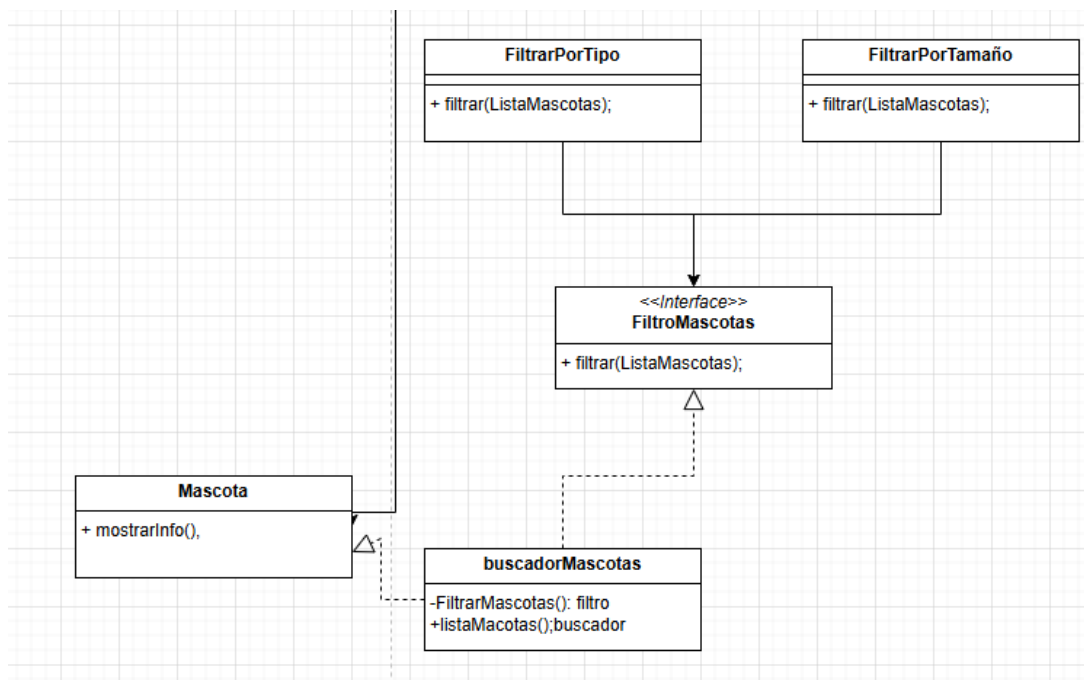
## 1. Patrón Strategy

**Aplicado en:** FiltrarPorTipo, FiltrarPorTamaño, BusquedaDeMascotas.

**Descripción:** Permite definir diferentes algoritmos de filtrado de mascotas y seleccionarlos dinámicamente según el criterio del usuario. Cada estrategia de filtrado se implementa en una clase separada.

**Ventajas:**

- Facilita la extensión con nuevos criterios de búsqueda.
- Reduce el acoplamiento entre la lógica de búsqueda y los filtros.



## 2. Patrón Factory Method

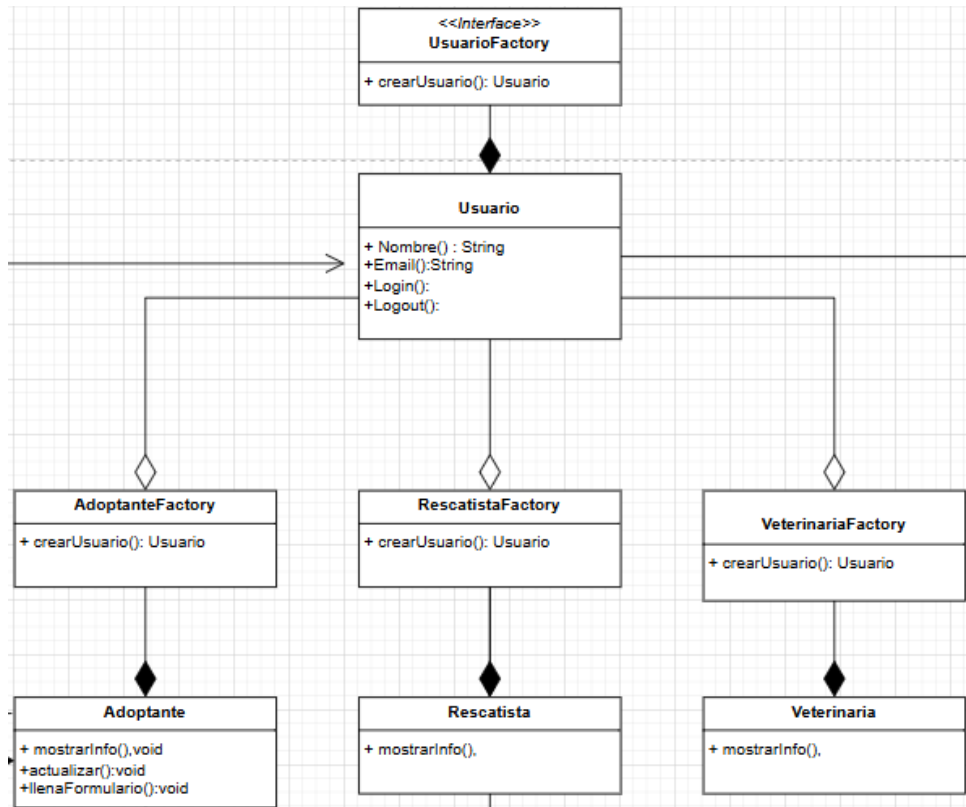
**Aplicado**

**en:** UsuarioFactory, AdoptanteFactory, RescatistaFactory, VeterinariaFactory.

**Descripción:** Este patrón permite encapsular la lógica de creación de objetos específicos según el tipo de usuario. En lugar de instanciar directamente las clases, se delega la creación a una fábrica, lo que facilita la extensión del sistema con nuevos tipos de usuarios sin modificar el código existente.

### Ventajas:

- Facilita la escalabilidad del sistema.
- Mejora la organización del código.
- Aplica el principio de abierto/cerrado.



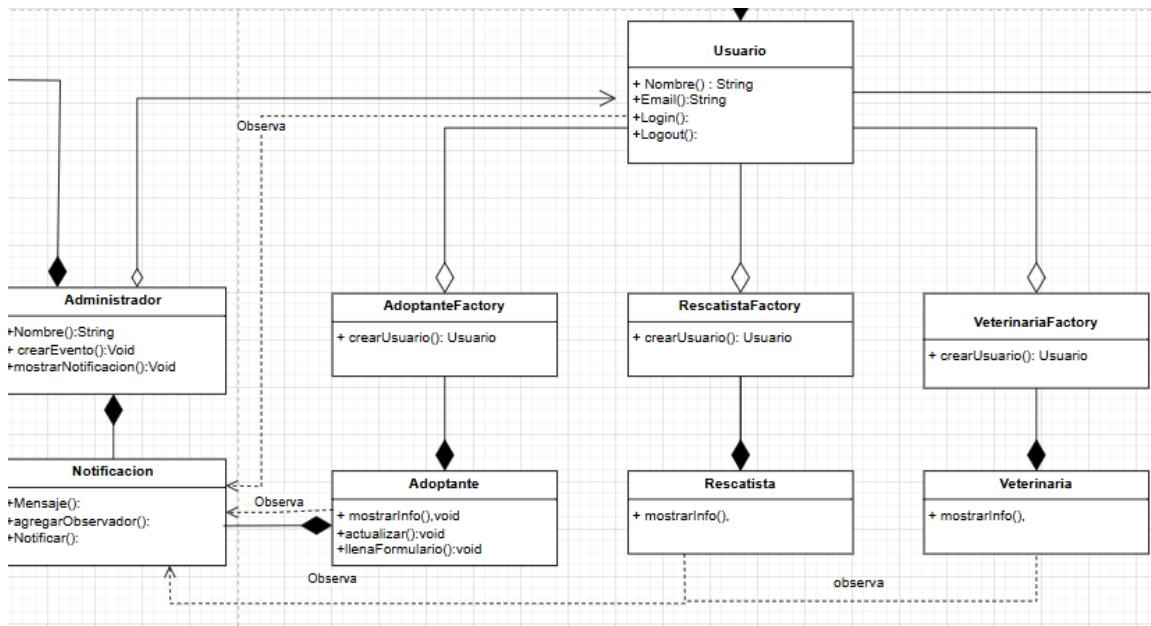
## 4. Patrón Observer

**Aplicado en:** Notificación.

**Descripción:** Permite que múltiples usuarios (como adoptantes o administradores) se suscriban a eventos del sistema (como nuevas mascotas disponibles o eventos de adopción) y reciban notificaciones automáticamente.

### Ventajas:

- Desacopla el emisor de eventos de los receptores.
- Facilita la implementación de notificaciones en tiempo real.



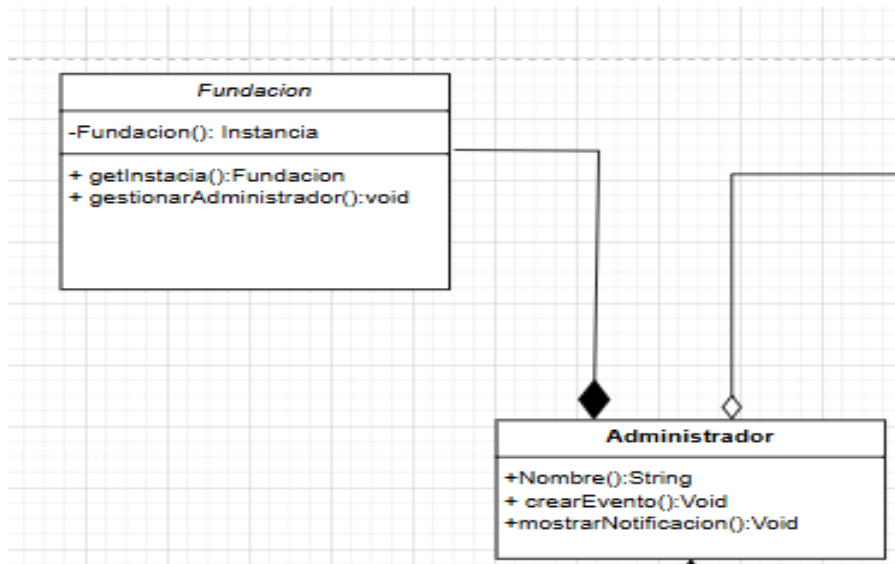
## 2. Patrón Singleton

**Aplicado en:** Fundación.

**Descripción:** Se asegura de que solo exista una única instancia de la clase Administrador y Fundación en todo el sistema. Esto es útil para centralizar la gestión de recursos, configuraciones o administración general.

**Ventajas:**

- Control global de acceso a la instancia.
- Evita inconsistencias en la gestión de la fundación

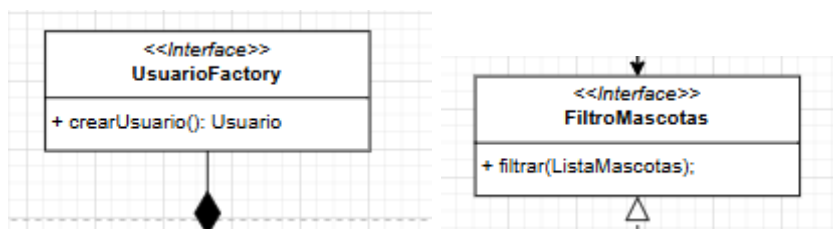


## 5. Patrón MVC (Modelo-Vista-Controlador) (*implícito en la estructura general*)

**Aplicado en:** Separación entre clases de datos (Usuario, Mascota, Formulario), lógica (Administrador, BusquedaDeMascotas) y presentación (no incluida en el diagrama, pero asumida en la interfaz).

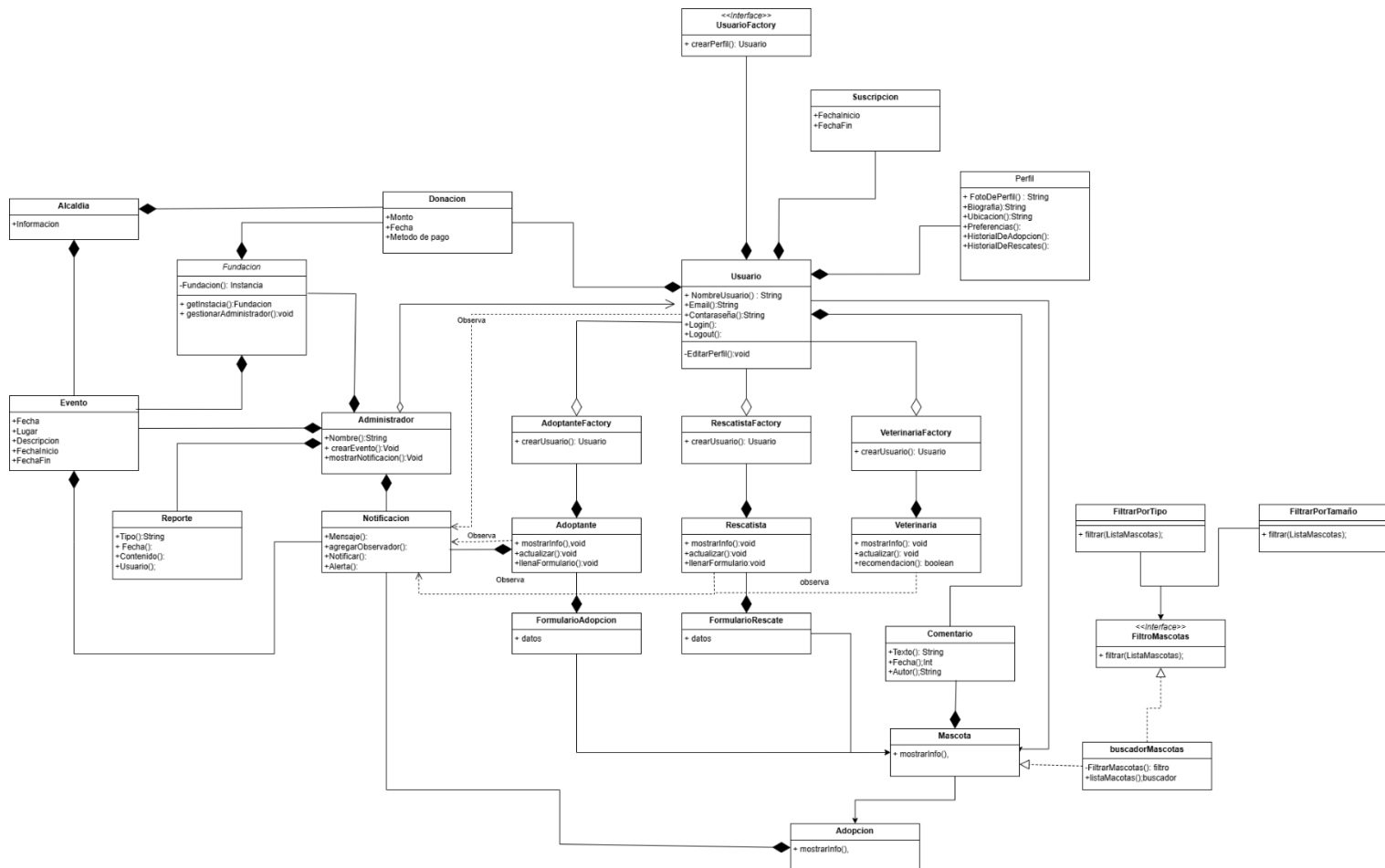
### Ventajas:

- Mejora la mantenibilidad del sistema.
- Permite que diferentes desarrolladores trabajen en capas separadas.
- Facilita pruebas y reutilización de componentes.



## Diagrama de Clases General

A continuación se presenta el diagrama de clases general del sistema:





## CONCLUSIONES

El diseño orientado a objetos del sistema "Rescatando Mascotas Forever" permite una representación clara y estructurada de los actores, procesos y entidades involucradas. La modularización facilita la comprensión y mantenimiento del sistema, y los diagramas de clases proporcionan una base sólida para el desarrollo del software. Se aplicaron buenas prácticas de diseño y se utilizaron herramientas TIC para la elaboración de los diagramas.

Al realizar este trabajo se comprendieron varios aspectos sobre el modelado en UML, destacando la importancia y como permite mejorar análisis de los casos dados.

Finalmente se aprendió a manejar herramientas para modelo de UML, algunas sugerencias y el diseño más claro y coherente.